ISSN 2710 - 1673 Artificial Intelligence 2024 № 3

УДК: 621.311 https://doi.org/10.15407/jai2024.03.104

О. Островська[1], В. Гура[2], Л. Добуляк[3]
[1,2,3]Львівський національний університет імені Івана Франка, Україна
1, вул. Університетська, м. Львів, 79000
[1]oksana.ostrovska@lnu.edu.ua
[2]volodymyr.gura@lnu.edu.ua
[3]lesia.dobuliak@lnu.edu.ua
[1]https://orcid.org/0009-0006-3376-8448
[2]https://orcid.org/0009-0007-8781-8970
[3]https://orcid.org/0000-0001-8665-8783

ВИКОРИСТАННЯ МОДЕЛЕЙ ГЛИБОКОГО НАВЧАННЯ ДЛЯ ВИЯВЛЕННЯ ШТУЧНО ЗГЕНЕРОВАНИХ ЗОБРАЖЕНЬ

Анотація. Використання моделей машинного навчання для класифікації реальних та штучно згенерованих зображень стає все більш актуальним напрямком досліджень у сфері штучного інтелекту. Ця стаття присвячена аналізу та розробці різноманітних моделей машинного навчання, що застосовуються для цієї задачі. Наша основна мета полягає в тому, щоб дослідити ефективність різних архітектур нейронних мереж у класифікації зображень, які були створені як людиною, так і за допомогою штучного інтелекту.

В рамках дослідження ми використовували дев'ять різних архітектур нейронних мереж, розроблених з метою аналізу їх ефективності в класифікації зображень. Ми вивчали як реальні зображення, так і ті, що були згенеровані за допомогою штучного інтелекту. Останні стають все більш поширеними в сучасному світі, а, отже, з'являється необхідність в їх визначенні та класифікації.

Основні результати нашого дослідження включають детальний аналіз характеристик згенерованих зображень, а також порівняння моделей за допомогою різних метрик. Ми використовували такі метрики, як точність та F1-оцінка, що дозволили нам об'єктивно оцінити роботу кожної з моделей. Крім того, ми визначили найбільш ефективні моделі для задачі класифікації зображень.

Одним із ключових висновків нашого дослідження є те, що моделі, які використовують регуляризацію та аугментацію даних, показали кращі результати. Регуляризація та аугментація даних допомагають забезпечити більш стабільну точність класифікації та зменшують схильність до перенавчання, що є важливою проблемою в галузі машинного навчання.

Ці результати можуть бути корисними для розробки стратегій протидії дезінформації, що стала проблемою в сучасному інформаційному суспільстві. Завдяки використанню моделей машинного навчання можна виявляти штучно згенеровані зображення та відокремлювати їх від реальних, що допоможе запобігти поширенню фальшивих новин та дезінформації.

Крім того, результати нашого дослідження можуть сприяти подальшому вдосконаленню алгоритмів генерації зображень, що є важливим напрямком розвитку штучного інтелекту. Вони можуть слугувати основою для створення більш точних та ефективних моделей, які зможуть генерувати зображення, що максимально наближені до реальних.

Ключові слова: машинне навчання, класифікація зображень, штучний інтелект, нейронні мережі, регуляризація, аугментація даних, дезінформація, алгоритми генерації зображень, точність, F1-оцінка, перенавчання.

O. Ostrovska[1], V. Hura[2], L. Dobulyak[3]
[1,2,3]Ivan Franko National University of Lviv, Ukraine
1, Universytetska str., Lviv, 79000
[1]oksana.ostrovska@lnu.edu.ua
[2]volodymyr.gura@lnu.edu.ua
[3]lesia.dobuliak@lnu.edu.ua
[1]https://orcid.org/0009-0006-3376-8448
[2]https://orcid.org/0009-0007-8781-8970
[3]https://orcid.org/0000-0001-8665-8783

USING MACHINE LEARNING MODELS TO DETECT ARTIFICIALLY GENERATED IMAGES

Abstract. The use of machine learning models to classify real and artificially generated images is becoming an increasingly relevant area of research in the field of artificial intelligence. This paper is dedicated to the analysis and development of various machine learning models used for this task. Our main goal is to investigate the effectiveness of

different neural network architectures in classifying images that have been created both in by human and with the help of artificial intelligence.

As part of the study, we used nine different neural network architectures designed to analyse their effectiveness in classifying images. We studied both real images and those generated by artificial intelligence. The latter are becoming more and more common in the modern world, which means that there is a need for their identification and classification.

The main results of our study include a detailed analysis of the characteristics of the generated images, as well as a comparison of the models using various metrics. We used metrics such as accuracy and F1-score, which allowed us to objectively evaluate the performance of each model. In addition, we identified the most effective models for the image classification task.

One of the key findings of our study is that models that use data regularisation and augmentation performed better. Data regularisation and augmentation help to ensure more stable classification accuracy and reduce the tendency to overfitting, which is an important issue in machine learning.

These results can be useful for developing strategies to counteract disinformation, which has become an issue in the modern information society. By using machine learning models, artificially generated images can be detected and separated from real images, which can help prevent the spread of fake news and disinformation.

In addition, the results of our research can contribute to the further improvement of image generation algorithms, which is an important area of artificial intelligence development. They can serve as a basis for creating more accurate and efficient models that can generate images that are as close to real life as possible.

Keywords: machine learning, image classification, artificial intelligence, neural networks, regularisation, data augmentation, misinformation, image generation algorithms, accuracy, F1 score, retraining.

Вступ

У сучасному світі, де штучний інтелект (ШІ) все більше впливає на різні аспекти нашого життя, виникає необхідність в розробці надійних технологій, які зможуть відрізнити згенеровані ШІ зображення від справжніх фотографій або малюнків. Ця наукова робота є спробою дослідити цю проблему, протестувати та проаналізувати різні методи та алгоритми, що допоможуть у створенні ефективної моделі для такої класифікації.

Особливий акцент у цьому дослідженні робиться на використання глибокого навчання та нейронних мереж. Ці методи вже успішно застосовуються в різних задачах класифікації зображень і демонструють високу ефективність, тому їх потенціал у контексті вирішення поставленої проблеми варто розглядати більш детально.

Модель, яка була розроблена в рамках цієї роботи, має великий потенціал для практичного застосування. Зокрема, вона може стати частиною систем автентифікації контенту в соціальних мережах, де вона допоможе виявляти та видаляти фальшивий контент на основі згенерованих зображень. Крім того, ця модель може бути корисною в наукових дослідженнях, де вона допоможе визначати автентичність зображень або виявляти зміни, внесені за допомогою ШІ.

Розробка та вдосконалення методів генерації зображень, в свою чергу, можуть відкрити нові можливості для їх застосування. Зокрема, більш точні та реалістичні згенеровані зображення можуть бути використані в таких сферах, як створення віртуальної реальності, тренування моделей штучного інтелекту, створення візуалізацій у науці та освіті, онлайн-реклама та інше.

Методи та засоби реалізації

Нейронні мережі - це системи обчислень, які були розроблені для імітації способу, яким людський мозок аналізує та обробляє інформацію. Вони складаються з великої кількості взаємопов'язаних обчислювальних вузлів, відомих як нейрони. Назва "нейронна мережа" походить від біологічних нейронів у мозку людини, які вони намагаються відтворити.

Кожен нейрон в нейронній мережі отримує набір вхідних даних, до яких застосовуються ваги. Ваги - це числові коефіцієнти, які визначають, наскільки важливий кожен вхід для вихідного сигналу нейрона. Потім до суми вагових входів додається зміщення, яке дозволяє нейрону регулювати свій вихід.

Робота нейронної мережі складається з двох основних етапів: прямого розповсюдження та зворотного розповсюдження. Пряме розповсюдження - це процес, що перетворює вхідні дані в вихідний сигнал. Вхідні дані проходять

через кожний шар мережі в одному напрямку, від вхідного шару до вихідного шару, де генерується кінцевий результат. Зворотне розповсюдження - це метод, за допомогою якого мережа вчиться, коригуючи свої ваги та зміщення на основі помилок у своїх прогнозах. Після генерації вихідного сигналу мережа порівнює свій прогноз зі справжнім значенням за допомогою функції втрат.

Згорткові нейронні мережі (Convolutional Neural Networks, CNN) - це клас глибоких нейронних мереж, які були спеціально розроблені для обробки зображень. Вони широко використовуються в різноманітних задачах комп'ютерного бачення, таких як класифікація зображень, розпізнавання об'єктів, семантична сегментація зображень тощо.

Основна особливість CNN полягає в тому, що вони використовують згортки (convolutions) замість звичайних повноз’єднаних шарів, які використовуються в більш традиційних глибоких нейронних мережах. Згортки дозволяють мережі автоматично та адаптивно вчити просторові ієрархії ознак з зображень, що забезпечує більш ефективне навчання.

Шари згорткових мереж

Згортковий шар (Convolutional Layer).

Згортковий шар є основним будівельним блоком CNN. Він сканує вхідне зображення за допомогою набору ваг, відомих як "ядро згортки" або "фільтр", та виконує операцію згортки, що дозволяє витягнути просторові ознаки з зображення.

Основні параметри:

– filters - кількість фільтрів в згортковому шарі. Кожний фільтр вчиться видобувати окрему ознаку з вхідних даних.

– kernel_size - розмір ядра згортки. Він визначає розмір вікна, яке "проходить" по вхідному зображенню.

– strides - крок, з яким вікно проходить по вхідному зображенню.

– padding - визначає, чи буде додано нулі до країв вектора вхідного зображення, щоб зберегти його розмір після згортки.

– activation - функція активації, яка застосовується після кожного шару. Найчастіше використовується ReLU (rectified linear unit).

Агрегувальний шар (Pooling Layer).

Агрегувальний шар використовується для зменшення просторового розміру вхідного зображення. Це допомагає зменшити кількість параметрів моделі і запобігає перенавчанню. Найчастіше використовується максимізаційне агрегування, яке вибирає максимальне значення з кожного вікна вхідних даних.

Основні параметри:

– pool_size - розмір вікна агрегації. Він визначає розмір вікна, яке "проходить" по вхідному зображенню.

– strides - крок, з яким вікно проходить по вхідному зображенню.

– padding - визначає, чи буде додано нулі до країв вектора вхідного зображення перед виконанням агрегації.

Повноз’єднаний шар (Fully Connected Layer, Dense).

Повноз’єднаний шар використовується в кінці мережі для виконання класифікації на основі вивчених ознак. Всі вузли в цьому шарі з'єднані з усіма вузлами в попередньому шарі.

Основні параметри:

– units - кількість вузлів в шарі. Вона відповідає кількості класів у задачі класифікації.

– activation - функції активації використовуються в нейронних мережах для внесення нелінійності в модель. Вони визначають вихід кожного нейрона на основі його входу.

В роботі використані такі найпопулярніші функції:

Сигмоїдна функція (Sigmoid):

$$f(x) = \frac{1}{1 + e^{-x}}.$$

Сигмоїдна функція конвертує будь-яке число в діапазон від 0 до 1, що робить її корисною для моделей, що прогнозують ймовірності. Проте, вона може страждати від проблеми "вимирання

градієнтів", коли градієнти стають дуже маленькими і мережа перестає навчатися.

ReLU (Rectified Linear Unit):

$$f(x) = max(0, x)$$

ReLU дуже проста функція, яка повертає 0 для від'ємних чисел і просто повертає x для додатних чисел. ReLU популярна через свою простоту та ефективність, але вона може страждати від проблеми "мертвого нейрона", коли нейрони перестають реагувати на будь-які вхідні дані.

Dropout Layer.

Dropout - це техніка регуляризації, що запобігає перенавчанню. Вона випадково вимикає вказану частку нейронів під час кожного проходу вперед, що допомагає забезпечити, що модель не стане занадто залежною від будь-якого окремого нейрона.

Flatten Layer.

Flatten шар використовується для перетворення багатовимірного вектора вхідних даних в одномірний вектор. Це зазвичай використовується перед повноз'єднаним шаром для підготовки даних.

Після аналізу архітектури та ефективності моделей, важливими компонентами є і інші параметри, які можуть впливати на результати моделей.

Оптимізатори.

Оптимізатори використовуються для оновлення ваг моделі на основі градієнтів, обчислених під час зворотного поширення помилки. Найпопулярнішим оптимізатором є Adam, який адаптивно налаштовує швидкість навчання для кожного параметра.

Функції втрат.

Функції втрат використовуються для обчислення помилки моделі під час навчання. Вибір функції втрат залежить від специфіки задачі. Для задач класифікації найчастіше використовуються Cross-Entropy Losses. Цей метод вимірює "відстань" між реальними та прогнозованими відповідями моделі.

Метрики.

Метрики використовуються для оцінки продуктивності моделі під час навчання та тестування. Для задач класифікації найчастіше використовуються точність (accuracy), повна відповідність (recall), точність (precision) та F1-оцінка (F1-score).

Кількість епох (epochs).

Епоха - це один повний прохід по всьому набору навчальних даних в процесі тренування. Вказівка кількості епох - це кількість разів, коли алгоритм пройде через весь набір даних. Якщо кількість епох занадто мала, модель може бути недонавченою, тобто вона може не встигнути вивчити всі шаблони в навчальних даних. Якщо ж кількість епох занадто велика, модель може перенавчитися, тобто занадто "зафіксуватися" на навчальних даних і втратити здатність до узагальнення.

Розмір пакета (batch size).

Розмір пакета визначає кількість прикладів навчання, які пропускаються через мережу перед тим, як оновлюються ваги.

Занадто малі пакети можуть призвести до нестабільного навчання та поганого узагальнення. Занадто великі пакети можуть бути неефективними з точки зору використання пам'яті та обчислювальних ресурсів. Крім того, вони можуть призвести до швидкої збіжності до мінімуму, який не є оптимальним (тобто до локального мінімуму, а не глобального).

Аугментація даних - це процес, при якому збільшується кількість даних шляхом створення модифікованих версій вже існуючих даних. Це допомагає покращити продуктивність моделей машинного навчання, оскільки вони отримують більш різноманітні дані для навчання. Аугментація зображень також дозволяє моделям бути менш чутливими до положення, орієнтації, освітлення та інших аспектів зображення, що допомагає покращити їхню продуктивність в реальному світі.

Можна виокремити такі методи аугментації зображень:

1. Обертання: зображення можна обертати на довільний кут. Це може допомогти моделі краще розпізнавати об'єкти, які знаходяться під різними кутами.

2. Зсув: зображення можуть бути зсунуті вгору, вниз, вліво або вправо. Це допомагає моделі краще розпізнавати об'єкти, які можуть не знаходитися в центрі зображення.

3. Масштабування: зображення можуть бути збільшені або зменшені. Це може допомогти моделі краще розпізнавати об'єкти різних розмірів.

4. Віддзеркалення: зображення можуть бути віддзеркалені по горизонталі або вертикалі. Це може допомогти моделі краще розпізнавати об'єкти, орієнтація яких може варіюватися.

5. Зміна яскравості: яскравість зображення може бути змінена. Це може допомогти моделі краще розпізнавати об'єкти при різних умовах освітлення.

6. Шум: в зображення може бути внесено деякий шум. Це може допомогти моделі стати більш стійкою до випадкового або небажаного зміщення у вхідних даних.

7. Деформація: зображення можуть бути деформовані шляхом стиснення або розтягування. Це може допомогти моделі краще розпізнавати об'єкти, форма яких може варіюватися.

Результати та їх аналіз

Для даної роботи були зібрані зображення з декількох наборів даних з платформи Kaggle загальним обсягом 2794 зображення, з них 1665 згенеровані.

На рис. 1 нижче приклад реальних фото, малюнків, графіків, а на рис. 2 приклад згенерованих фото.



Рис. 1. Приклади реальних зображень



Рис. 2. Приклади згенерованих зображень

Спочатку знайдемо стандартне відхилення у кожному класі даних.

$$\sigma = \sqrt{\frac{(\Sigma(x_i - \mu)^2)}{N}},$$

де:

– x_i - кожне окреме значення в наборі даних

– μ - середнє значення набору даних

– N - кількість значень в наборі даних.

Згенеровані зображення мають вище середнє стандартне відхилення (65.29), ніж

реальні зображення (63.01). Це може свідчити про те, що згенеровані зображення в середньому мають більше варіативності або "шуму" порівняно з реальними зображеннями.

Також знайдемо середні значення RGB. Середні значення RGB для згенерованих зображень є нижчими, ніж для реальних зображень. Це свідчить про те, що ШІ-згенеровані зображення в середньому є темнішими або менш насиченими, ніж реальні зображення.

Кореляційна матриця для набору даних обчислюється за формулою:

$$corr(X,Y) = \sum \frac{(x_i - \mu_x)(y_i - \mu_y)}{(N - 1)\sigma_x \sigma_y},$$

де:

– X, Y - це два виміри (або два ряди) в даних.

– x_i, y_i - це окремі значення в рядах X і Y відповідно.

– μ_x, μ_y - це середні значення рядів X і Y відповідно.

– σ_x, σ_y - це стандартні відхилення рядів X і Y відповідно.

– N - це кількість значень в кожному ряду.

```
[1.        , 0.90133621, 0.77827218]
[0.90133621, 1.        , 0.89511924]
[0.77827218, 0.89511924, 1.        ]
```

Рис. 3. Матриця кореляції для реальних зображень

```
[1.        , 0.86893171, 0.71370172]
[0.86893171, 1.        , 0.85859611]
[0.71370172, 0.85859611, 1.        ]
```

Рис. 4. Матриця кореляцій для згенерованих зображень

Матриці кореляції (рис. 3 та рис. 4) для обох типів зображень показують високу позитивну кореляцію між каналами RGB. Проте, кореляція між каналами є вищою для реальних зображень, особливо між каналами R та B. Це може свідчити про те, що колірні складові в реальних зображеннях є більш тісно пов'язаними, ніж в згенерованих зображеннях.

Перед поданням даних в модель потрібно стандартизувати розмір зображення та нормалізувати їх.

Зменшимо розмір усіх зображень до 64x64, що дозволить швидко опрацьовувати їх та зменшить шуми.

Нормалізуємо зображення. Поділимо значення пікселів на 255, що є максимальним можливим значенням пікселя в 8-бітному зображенні та перетворює значення пікселів до діапазону від 0 до 1.

Об’єднаємо зображення в один набір даних та їх класи в інший. Розділимо зображення на набір для тренування та тестування моделей за принципом 80%/20%.

Для оцінювання і порівняння ефективності моделей використовуватимемо такі метрики як точність (accuracy) та F1-оцінка.

$$Точність = (TP + TN) / (TP + TN + FP + FN)$$

де:

– *TP* - це кількість True Positives (справжні позитивні), тобто кількість позитивних випадків, які правильно класифіковані як позитивні.

– *TN* - це кількість True Negatives (справжні негативні), тобто кількість негативних випадків, які правильно класифіковані як негативні.

– *FP* - це кількість False Positives (хибні позитивні), тобто кількість негативних випадків, які помилково класифіковані як позитивні.

– *FN* - це кількість False Negatives (хибні негативні), тобто кількість позитивних випадків, які помилково класифіковані як негативні.

$$F1 = 2 * (Точність * Повнота) / (Точність + Повнота)$$

де:

– *Точність = TP / (TP + FP)*
– *Повнота = TP / (TP + FN)*

F1-оцінка враховує як точність, так і повноту, і є корисною метрикою для незбалансованих класів, оскільки вона штрафує моделі, які виконують погано в одному або обох цих аспектах.

У цьому дослідженні ми створили 9 моделей машинного навчання з різною архітектурою шарів, кожна з яких була навчена протягом 30-ти епох з використанням оптимізатора Adam.

– Модель 1: ця модель складається із згорткового шару, шару максимального згруповування, шару розгортки та повноз'єднаного шару. Згортковий шар має 32 фільтри розміром 3x3 і загалом 896 параметрів, що допомагає виявляти різні особливості зображення. Шар максимального згруповування зменшує розмір вихідних даних вдвічі, що прискорює обчислення. Заключний повноз'єднаний шар з одним виходом має 30,753 параметри, що дозволяє моделі робити прогнози.

– Модель 2: ця модель має два згорткові шари та два шари максимального згруповування. Другий згортковий шар має 64 фільтри розміром 3x3 і містить 18,496 параметрів. Це дозволяє моделі виявляти більш складні особливості зображення.

– Модель 3: ця згорткова модель з трьома згортковими шарами та одним повноз’єднаним шаром. Третій згортковий шар має 128 фільтрів, що дозволяє моделі виявляти ще більш детальні особливості зображення.

– Модель 4: ця модель має один згортковий шар, один шар регуляризації та один повноз’єднаний шар. Шар регуляризації допомагає запобігти перенавчанню моделі.

– Модель 5: складається з одного згорткового шару, одного шару нормалізації та одного повноз’єднаного шару. Шар нормалізації допомагає прискорити навчання моделі та зробити його більш стабільним.

– Модель 6: має два згорткові шари, один шар регуляризації, один шар нормалізації та один повноз’єднаний шар. Комбінація регуляризації та нормалізації допомагає моделі навчитися ефективніше та запобігти перенавчанню.

– Модель 7: складається з одного згорткового шару та одного повноз’єднаного шару з більшою кількістю нейронів. Збільшення кількості нейронів допомагає моделі виявляти більше шаблонів у даних.

– Модель 8: використовує аугментацію даних та має просту архітектуру. Аугментація даних допомагає моделі краще досягти узагальнення, створюючи додаткові варіації даних для навчання.

– Модель 9: має вдвічі більшу кількість пакетів та просту архітектуру. Збільшення кількості пакетів допомагає моделі краще навчитися на даних.

Кожна модель оцінена за її точністю, швидкістю навчання та здатністю уникнути перенавчання. Результати показали, що моделі з регуляризацією та аугментацією даних виявилися найефективнішими. Для кращого розуміння ефективності кожної моделі створено порівняльну таблицю 1, де представлені основні характеристики та результати кожної моделі. В таблиці можна побачити ключові метрики ефективності, такі як точність та F1 Score.

На рис. 5 представлено точність для навчального та тестового наборів даних для кожної з моделей.

Цей графік дозволяє нам візуально порівняти ефективність моделей на різних етапах навчання. Він також демонструє, як моделі адаптуються до даних протягом

часу та як їхня точність змінюється з кожною епохою.

Таблиця 1. Точність для тренувального та тестового набору даних

	Training Accuracy	Validation Accuracy	F1 Score
Model 1	0.998658	0.751342	0.767516
Model 2	1	0.735242	0.749196
Model 3	1	0.762075	0.770416
Model 4	0.969128	0.754919	0.785933
Model 5	1	0.73703	0.722853
Model 6	0.963311	0.783542	0.795597
Model 7	1	0.744186	0.735849
Model 8	0.740492	0.742397	0.773936
Model 9	0.94094	0.758497	0.784431

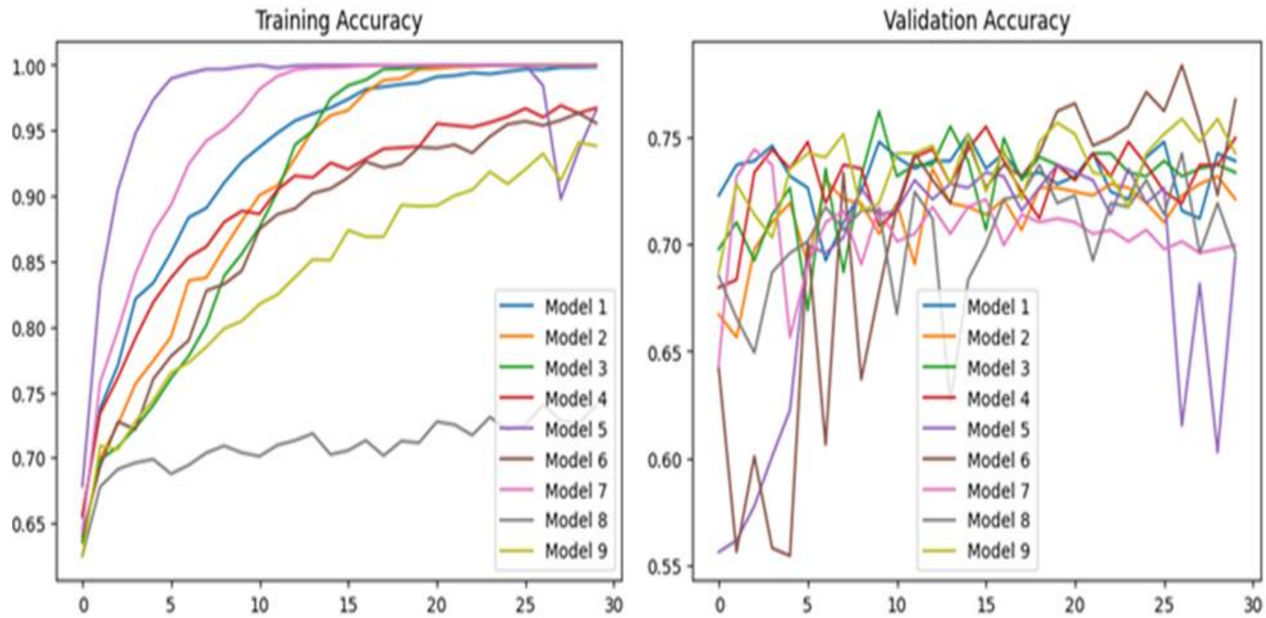


Рис. 5. Точність для тренувального та тестового набору даних

З даних графіків і таблиці можна зробити такий висновок:

Моделі 4, 6, 8, 9 показали найкращий баланс, та не досягали перенавчання. А це тому, що 4, 6 і 9 використовували регуляризацію (Dropout).

Модель 8 використовувала аугментацію даних, що призвело до того, що точність під час тренування була дуже близькою до точності під час валідації.

Модель 6, що дала найкращий результат під час перших епох, мала найгіршу точність валідації серед усіх моделей та повільно, але стабільно зростала.

Модель 5 з нормалізацією пакетів найшвидше досягла перенавчання, а саме за 5 епох, мала одну з найнижчих точностей під час валідації та найгірший показник F1. А після 25 епох точність валідації знизилася від 70% до 60%.

Більша кількість нейронів в шарі Dense в моделі 7 також призвела до швидкого перенавчання та поганих показників під час валідації.

Висновки

В ході виконання даної роботи було проведено дослідження та розробка моделей машинного навчання для класифікації зображень, зокрема для виявлення зображень, згенерованих штучним інтелектом (ШІ). Першочерговою метою було визначення, чи модель правильно класифікує зображення як реальні або згенеровані.

Під час аналізу вхідних даних було з'ясовано, що згенеровані зображення в середньому є темніші та містять більше шуму в порівнянні з реальними зображеннями. Ця інформація була корисною під час розробки моделей машинного навчання.

Для обробки даних було використано стандартизацію розміру зображення та нормалізацію значень пікселів, що допомогли забезпечити оптимальні умови для навчання моделей.

Було розроблено 9 моделей нейронних мереж з різною архітектурою. Це дозволило провести порівняльний аналіз та визначити, які моделі найкраще справляються з поставленою задачею.

При порівнянні моделей було з'ясовано, що деякі з них показали ознаки перенавчання, оскільки вони досягли 100% точності на тренувальних даних при значно нижчій точності на тестових даних. Це свідчило про те, що ці моделі вивчили специфічні особливості тренувального набору даних замість загальних шаблонів, які можна застосувати до нових даних.

Чотири з дев’яти моделей показали найкращий баланс між точністю навчання та валідації. Зокрема, модель з нормалізацією пакетів показала найвищу точність валідації та F1-оцінку, що робить її найкращим кандидатом.

Практичне застосування цієї роботи може охоплювати різні області. Вона може бути використана для виявлення згенерованих зображень в соціальних медіа або на веб-сайтах, що допоможе боротися з дезінформацією та соціальною інженерією. Також ця робота може бути корисна для розробників алгоритмів, дозволяючи їм оцінювати якість генерованих зображень, порівнювати різні алгоритми генерації зображень та визначати, які аспекти зображення є найскладнішими для відтворення для ШІ.

Проведене дослідження демонструє, як можна використовувати машинне навчання для вирішення конкретних задач, зокрема, для класифікації зображень. Результати показують, що згенеровані зображення непросто відрізнити, оскільки точність різних параметрів не досягала 80% чи вище.

Література

1. Набори даних. Kaggle. URL:https://www.kaggle.com/datasets (дата звернення: 02.09.2024).
2. Посібник з конволюційних нейронних мереж. TensorFlow. URL:https://www.tensorflow.org/tutorials/images/cnn (дата звернення: 02.09.2024).
3. Стаття про дослідження в галузі нейронних мереж. Arxiv. URL:https://arxiv.org/abs/1511.08458 (дата звернення: 02.09.2024).
4. Стаття про великі дані. Journal of Big Data. URL:https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00444-8 (дата звернення: 02.09.2024).
5. Повний посібник по конволюційним нейронним мережам. Towards Data Science. URL:https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53 (дата звернення: 02.09.2024).
6. Стаття про конволюційні нейронні мережі. Analytics Vidhya. URL:https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/ (дата звернення: 02.09.2024).
7. Вступ до конволюційних нейронних мереж. Geeks for Geeks. URL:https://www.geeksforgeeks.org/introduction-convolution-neural-network/ (дата звернення: 02.09.2024).

References

1. Data sets. Kaggle. URL: https://www.kaggle.com/datasets (accessed 02.09.2024).
2. Guide to convolutional neural networks. TensorFlow. URL: https://www.tensorflow.org/tutorials/images/cnn (accessed 02.09.2024).
3. An article about research in the field of neural networks. Arxiv. URL: https://arxiv.org/abs/1511.08458 (accessed 02.09.2024).
4. Article about big data. Journal of Big Data. URL: https://journalofbigdata.springeropen.com/article/. 10.1186/s40537-021-00444-8 (accessed: 02.09.2024).
5. A complete guide to convolutional neural networks. Towards Data Science. URL:https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53 (accessed: 02.09.2024).
6. Article about convolutional neural networks. Analytics Vidhya.

URL:ttps://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/
(accessed 02.09.2024).
7. Introduction to convolutional neural networks. Geeks for Geeks.
URL:https://www.geeksforgeeks.org/introduction-convolution-neural-network/
(accessed 02.09.2014): 02.09.2024).

The article has been sent to the editors 12.08.24.
After processing 14.09.24.
Submitted for printing 30.09.24.